Cluster-based Intrusion Detection (CBID) Architecture for Mobile Ad Hoc Networks

Ejaz Ahmed, Kashan Samad, Waqar Mahmood
NUST Institute of Information Technology (NIIT), Rawalpindi, Pakistan
{ ejaz, kashan.samad,, drwaqar}@niit.edu.pk

Abstract

The ad hoc networks are vulnerable to attacks due to distributed nature and lack of infrastructure. Intrusion detection systems (IDS) provide audit and monitoring capabilities that offer the local security to a node and help to perceive the specific trust level of other nodes. The clustering protocols can be taken as an additional advantage in these processing constrained networks to collaboratively detect intrusions with less power usage and minimal overhead. Existing clustering protocols are not suitable for intrusion detection purposes, because they are linked with the routes. The route establishment and route renewal affects the clusters and as a consequence, the processing and traffic overhead increases due to instability of clusters. The ad hoc networks are battery and power constraint, and therefore a trusted monitoring node should be available to detect and respond against intrusions in time. This can be achieved only if the clusters are stable for a long period of time. If the clusters are regularly changed due to routes, the intrusion detection will not prove to be effective. Therefore, a generalized clustering algorithm has been proposed that can run on top of any routing protocol and can monitor the intrusions constantly irrespective of the routes. The proposed simplified clustering scheme has been used to detect intrusions, resulting in high detection rates and low processing and memory overhead irrespective of the routes, connections, traffic types and mobility of nodes in the network. Clustering is also useful to detect intrusions collaboratively since an individual node can neither detect the malicious node alone nor it can take action against that node on its own.

1. Introduction

Mobile Ad Hoc Networks, shortly called MANETs, is one of the major research domains. They have undergone rapid growth in the past several years because of their application in military and rescue services, battlefield and disaster recovery operations, mobile conferencing and variety of other applications. Their dynamic nature makes these systems susceptible to various attacks.

Ad hoc networks do not have any fixed topology. Each node not only acts as a host, but also as a packet forwarding entity such as a router. As opposed to infrastructure based wireless networks, the ad hoc networks do not require a base station. The nodes can move freely in any direction and they can join or leave the network at any time. Besides being advantageous of having low deployment cost, these networks are battery and power constrained. The nodes in an ad hoc network can vary from a Laptop to a Cell Phone. These devices have limited power and processing capabilities. Therefore, the more the network grows, the more they are required to forward packets for other nodes; devoting a significant amount of processing power. The other major challenges of adhoc networks are their dynamic topology and limited bandwidth.

Due to dynamic nature and lack of centralized monitoring points, the adhoc networks are vulnerable to various kinds of attacks. They suffer from the vulnerabilities that arise in wired communications like passive eavesdropping, spoofing, denial of service, access control, authorization, etc [1]. They also experience vulnerabilities due to wireless nature of the network, like blackhole, wormhole, sinkhole, selfishness, sleep deprivation, etc. The ad hoc routing protocols also contains deficiencies and they may suffer from attacks

such as rushing and fabrication etc. One of the major problems in adhoc network is that all the nodes are trusted equally and therefore, the intermediate node in the route can easily fabricate or drop the control or data packets. In short, the distributed infrastructure-less nature of MANETs make intrusion detection a challenge.

Intrusion Detection System (IDS) can be deployed in these self-organizing multi-hop wireless networks to protect them against a number of attacks by offering auditing and monitoring capabilities to a node. However, normal intrusion detection approaches cannot be used in this environment. Since these networks lack infrastructure, we need to monitor intrusions at all nodes in the network. But, due to mobility and other constraints such as restricted power and processing capacity, nodes cannot run heavy applications to detect intrusions. If every node starts monitoring intrusions separately, processing overhead at each node will consume a large portion of their battery and power. Therefore, a scalable and fault tolerant IDS is required to govern these non-secure wireless adhoc networks against attacks.

The Intrusion Detection architecture should be simple yet effective to provide security against different type of attacks. The efficient solution is to defend against intrusion co-operatively, rather than each mobile node performing full analysis of traffic passing through it. In order to co-operate, the nodes must trust each other so that they don't have to audit all the data, saving a lot more processing and memory overhead. The clustering in MANETS can be taken as an advantage in these battery and memory constrained networks for the purpose of intrusion detection, by separating tasks for the head and member nodes, at the same time providing opportunity for launching collaborative intrusion detection. The clustering schemes are generally used for the routing purposes to enhance the route efficiency. However, the effect of change of a cluster tends to change the route; thus degrades the performance. Therefore, a low-overhead clustering algorithm is proposed in [5] for the benefit of detecting intrusion rather than efficient routing. The proposed simplified clustering scheme is used to detect intrusions under various attacks such as blackhole, routing loop, selfishness, and sleep deprivation in Mobile Adhoc Networks environment. The results show that the architecture is simple in terms of clustering and election process, and effective in terms of intrusion detection and response.

The rest of the paper is organized as follows: Section 2 outlines the related work, section 3 outlines the clustering concepts for ad-hoc networks. Section 4 discusses the cluster formation algorithm. In section 5, the intrusion detection framework is being discussed using the proposed clustering scheme. Section 6 contains the conclusion and future work is provided in section 7.

2. RELATED WORK

Mobile Adhoc Network security is addressed by various researches and has been a major research area. Intrusion detection in Adhoc networks by using modular approach is proposed by Zhang *et al.* [2]. In this proposed solution every node is responsible to detect the intrusion independently. The nodes can collaborate in scenarios where an individual node can not conclude about the intrusive behaviour. In this architecture each node runs various modules including local, global detection engine and response. A complex multilayer integration approach is used to analyse the intrusion which results in storing lot of information on each node thus making it storage and processing intensive.

Using mobile agents (MA) in intrusion detection is a new dimension in Adhoc Network Security research. Li *et al.* [3] proposed a coordinated approach of intrusion detection in ad-hoc networks using MA

technology. The Manager, assistant and response mobile agents are used for detection and notification of intrusion within a network. The proposed architecture floods the network with intrusion information thus resulting in processing and storage overhead on each individual node.

Yi-an *et al.* [4] proposed a cooperative intrusion detection system for MANETS. The run-time resource constraint problem was addressed using a cluster-based intrusion detection scheme. The cluster formation and cluster head selection for cooperative intrusion detection is done through clique computation and cluster head computation protocols. The requirement of having bi-directional links for clique computation protocol causes an overhead in terms of number of elections and number of HELLO messages exchanged to maintain connectivity.

Apart from the secure architectures various clustering algorithms have been proposed for Mobile Adhoc Networks for efficient routing, as their dynamic nature makes routing a difficult task. Some of these algorithms are described in [6, 7, 8].

3. Cluster Formation

The clusters are formed to divide the network into manageable entities for efficient monitoring and low processing in the network. The clustering schemes result in a special type of node, called the "Head Node" (HD) to monitor traffic within its cluster. It not only manages its own cluster, but also communicates with other clusters for cooperative detection and response. It maintains information of every member node and neighbor clusters, which is useful for network-wide communication. The cluster management responsibility is rotated among the capable members of the cluster for load balancing and fault tolerance [6] and must be fair and secure [4]. This can be achieved by conducting regular elections. The proposed election process [5] is simple. It does not require the clique computation [4], or the neighbor information [7]. The cluster-head keeps an election interval timer for managing the elections. Every node in the cluster must participate in the election process by casting their vote showing their willingness to become the cluster-head. The node showing the highest willingness (or proves the best following some criteria) becomes the cluster-head until the next timeout period.

The clustering algorithm was presented in detail in [5] containing discussion about node states, data structures, HELLO messages, cluster-head nomination (election) process, and verification of votes and results. The following analysis has shown that the proposed ID clustering scheme [5] results in lesser clusters when compared with scheme presented in [4]. For comparative analysis the 18 node base topology has been taken from [8]. It is observed that clustering scheme proposed in [4] results in 9 clusters as shown in figure 1b and 5 clusters by using scheme proposed in [5] as shown in figure 1c.

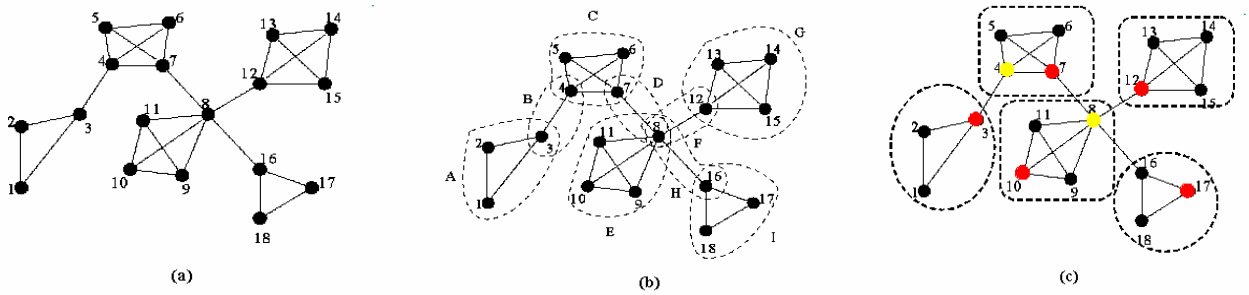

Figure 1. Cluster formation using various algorithms in 18-node topology
(Head Nodes are shown in Red, Gateway Nodes in Yellow and Member Nodes in Black)

4. Intrusion Detection Architecture

An IDS is used to detect attempted intrusion into a computer or network. It processes audit data, performs analysis and takes certain set of actions against the intruder, such as blocking them and/or informing the system administrator. Ad hoc networks lacks in centralized audit points [9,10], therefore, it is necessary to use the IDS in a distributed manner. This also helps in reducing computation and memory overhead on each node.

The proposed clustering algorithm [5] can be related with the intrusion detection process as partial analysis of the incoming traffic is done at the head node and rest of the analysis is done at the intermediary or destination member node. The traffic analysis at head node and packet analysis at member node [4] is helpful in reducing processing at each node. If some malicious activity is found by HD, it informs its members and the neighboring clusters to take certain set of actions. It is the responsibility of cluster-head to obtain help from and/or inform the member nodes and neighboring clusters for a particular intrusion. It is important to note that a undecided node (UD) node performs its own audit and analysis; however, it performs partial analysis immediately after becoming head node (HD) or member node (MB).

IDS can be either host-based or network-based, depending on the monitoring level required. The techniques to detect intrusion can be anomaly detection or misuse/signature detection. The host-based IDS (HIDS) observe traffic at individual hosts, while network-based IDS (NIDS) are often located at various points along the network. Since centralized audit points are not available in ad hoc networks, we cannot use NIDS technique. Alternatively, if every host starts monitoring the intrusions individually such as in HIDS, lot of memory and processing will be involved. Therefore, a distributed and combined technique is used to perform effective monitoring in the network, where both the head and member nodes are involved in collecting audit data.

The IDS can be categorized as misuse detection system or anomaly detection system. Misuse detection (or signature detection) system is generally used for known patterns of unauthorized behavior (or attack signatures). The anomaly detection system identifies intrusions using ‘normal’ activity baseline. It achieves this with ‘self-learning’ [11]. The misuse detection system often fails if the database of attack signatures is not up to date. The other problem with misuse detection system is the bulk of database which an ad hoc node cannot handle due to memory constraint, if it contains all the known suspicious signatures. Therefore, anomaly detection technique is used that is trained with passage of time for normal traffic and this information is then further used in the testing period to detect abnormal activities/behavior.

A flow model of intrusion detection architecture of CBID [5] is presented in figure.2, which consists of 4 modules. These modules are linked with each other for effective intrusion detection. The information collected during the training phase in the logging module is passed regularly to the intrusion information module to perceive a threshold value for the normal traffic. This threshold value is further used for the traffic during the testing phase to check intrusive activity. If some abnormal behavior is found, an alert is generated by the intrusion response module. The functionality of each module is given below:

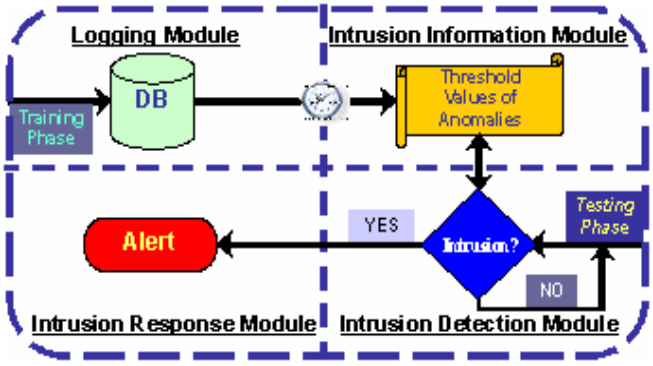

Figure 2. Intrusion Detection Process Flow

4.1. Logging Module

HD node logs all the traffic transferred through its radio range. It captures all the traffic in the promiscuous mode and keeps the necessary fields in a database. It keeps the data related to traffic such as number of packets sent, received, forwarded or dropped. The traffic can either be data traffic or the control traffic. The control traffic includes RREQ, RREP, RERR packets of AODV and HELLO and ELECTION packets of CBID. If keeps the count of packets transacted for each sampling interval. These logs can be helpful for detection of many attacks such as blackhole, wormhole, sleep deprivation, malicious flooding, packet dropping, etc.

The member nodes (MB) and gateway nodes (GW), log the route details, such as the number of routes added, removed, etc. These features can be helpful in detecting attacks like denial of service(SYN flooding) and route fabrication attacks.

4.2. Intrusion Information Module

If misuse signature technique is used, every node must maintain a database such as “intrusion interpretation base”, which includes the process of “learning” [3]. All the signatures that cause an intrusion must be kept in the database. For anomaly detection technique, the anomalous behaviors must also be well defined with proper upper and lower threshold values. The detection process may be used for either misuse signature or anomaly behavior (to conserve power and battery life) or both of them (to perform conclusive and efficient detection). The signature database or anomalous values can be updated manually or on the base of response from other network nodes.

The logging module values are used to perceive upper and lower threshold values for the anomalies. Mean and Standard Deviation Model [12] is one of the methods to process the data and measure the spread of normal traffic. Mean and standard deviation is calculated for each sample of data and to perceive the upper and lower threshold values the following formula is used:

$$mean + d * stdev$$

4.3. Intrusion Detection Module

When the nodes are trained, they detect the intrusions by analyzing and comparing the traffic patterns with the normal behavior. The HD node still captures the traffic in the promiscuous mode and compares its behavior with the normal traffic behavior. If anomaly is found in the data, the HD node raises the alarm, and increases the monitoring level and analyzes the traffic in more detail to find out the attack type and identity of the attacker. To preserve the resources, the HD node initially log only a few details of the traffic such as packet count. When an anomaly is found, the packet monitoring level can be increased such as analyzing the packet in depth depending on the resources available. If the intruder does not belong to the same cluster in which the suspicious behavior is detected, the HD node may ask neighbor cluster-heads to cooperate. This is the reason on maintaining neighbor cluster information by HD nodes in Cluster Member Table (CMT) [5] which is forwarded by the GW nodes.

4.4. Intrusion Response Module

To inform other nodes about some intrusion, head and member nodes generate alerts. The response may be local to the cluster or global, covering the whole network. When a member node detects an intrusion without any help from cluster-head, it takes "self-response" (e.g. blocking the current user) and informs the cluster-head about the intrusion. The cluster-head logs the entry and informs other nodes about the intrusive activity.

The cluster-head generates a "cluster-based response" to the cluster in any of the 3 cases: a member node has informed about an intrusion, after log-based detection or after getting response from adjacent cluster. The cluster-head can also generate a "network-wide response". In the first 2 cases of cluster-based response, network-wide response is optional, whereas in the third case, it is necessary to inform the whole network about the intrusion. For the network-wide response, the HD node generates a response and forwards it to its GW node which passes it to all its HD nodes. Those HD nodes than, generate a cluster-based response and forward the same response to their adjacent clusters. In this case, all the nodes in the network are informed about a misbehaving node, or some fabricated route message.

The response taken due to found intrusion can be one of the following: removing the malicious node 'M' from the route, reducing trust level of node M or blocking all the traffic from node M, etc. The trust level reduction can be helpful as preemptive measure so that:

- node M is not elected as a head node (HD) in the future, or
- no route involving node M is entertained.

5. Performance Evaluation

For performance evaluation a simulation is conducted in NS-2 using AODV as the underlying routing protocol. The simulation parameters are listed in table 2. The nodes move to a random direction with a randomly selected mobility speed ranging from 5 to 40 m/s. The simulation has been tested for both UDP and TCP traffic types. The traffic load is also varied with low, medium and high traffic load conditions.

Parameter	Value/Choice
Number Of Nodes	18
Topology Grid	1000m * 1000m
Training Time	10000s

Testing Time	1000s
Feature Sampling Interval	5s
Node Movement Model	Random Way-Point Model
Peak Movement Model	5 ~ 40 m/s
Transmission Range	250m
Maximum Bandwidth	1 Mb/s
Total Connections	10 (Maximum)

Table 2: Simulation Parameters

The features given in [4] are being used to test the simulated attacks. The traffic related features, shown in table 3, are calculated by each head node within a cluster.

Dimension	**Values**
Packet Type	Data, ROUTE REQUEST, ROUTE REPLY, ROUTE ERROR and HELLO messages
Flow Direction	sent, forwarded, received and dropped
Sampling Periods	5, 60 and 900 seconds
Static Measures	count and standard deviation of inter-packet intervals

Table 3: Traffic Related Features

The route related features, given in table 4, are intended for the member nodes within a cluster. It is assumed that the intrusion can only be launched on the data or route messages.

Dimension	**Values**
route add count	Routes newly added via route discovery
route removal count	stale routes being removed
route find count	routes in cache with no need to re-discovery
route notice count	routes added via overhearing
route repair count	broken routes currenlty under repair
total route change	route change rate within the period
average route length	average length of active routes

Table 4: Route Related Features

The attacks simulated are taken from [4, 13]. Following is a short description of these attacks:

1) *Blackhole and Sleep Deprivation using False Source Route and Maximum Sequence and Rushing*: The attacker advertises a false route from a victim node X, for any destination node D with a hop count set to 1. Since the victim node X does not have the legal route to the destination, it becomes blackhole as all the traffic is directed to X.

2) *Selfishness and Denial-of-Service using Packet Dropping*: A malicious node in the path suddenly starts dropping packets, causing other nodes a denial-of-service.

3) *Sleep Deprivation using Malicious Flooding*: A victim is flooded with large number of malicious packets by the intruder causing it to go in sleep deprivation due to battery drain.

4) *Routing Loop using Spoofing*: A routing loop is created by the intruder by falsifying route replies in response to the legitimate route requests from the nodes.

Each attack is carried out for 1/4th time of the testing phase by creating 5 connections each of 50 seconds of each attack at random time. For each sample, the values are compared with the threshold value learned by the nodes in the training phase. The CBID attack detection rate is compared with other intrusion detection systems. The results of CIDS and MLAD schemes are taken from the papers in [4,13].

Attack	**CIDS**	**MLAD**	**CBID**
Blackhole	85%	83.33%	91%
Selfishness	98%	72%	100%
Sleep Deprivation	99%	100%	100%
Routing Loop	87%	X	99%

Table 5: Successful Anomaly Detection rate

The above table describes the successful attack detection rate. It is noted that the proposed CBID architecture performs better in all the four simulated attacks when compared with CIDS and MLAD techniques.

.

Attack	**CIDS**	**MLAD**	**CBID**
Blackhole	0.97%	0.29%	0.53%
Selfishness	0.89%	0.29%	0.26%
Sleep Deprivation	0.95%	0.29%	0.40%
Routing Loop	0.98%	X	10.06%

Table 6: False Alarm Rate

The above table describes the false alarm rate for the three intrusion detection schemes. The overhead incurred by the cluster based intrusion detection scheme (CBID) under different traffic and load conditions at varying mobility situations are also measured. AODV has been used as the underlying routing protocol and UDP and TCP traffic is simulated to find the total number of packets i.e. AODV & UDP/TCP packets, transacted in the network.. The results depicts uniform overhead of CBID clustering scheme irrespective of the load and mobility conditions as shown in figure 3 to figure 8. It is observed that the number of packets transacted in the network at high mobility are decreased as compared to the low mobility. It is because either due to mobility, the destination node may come closer to the sender node and the number of hops transacted by the packet will reduce; thus reducing the number of forwarded packets, or due to non-availability of a route after the node moved to a position where it has no direct or indirect link with the sender node.

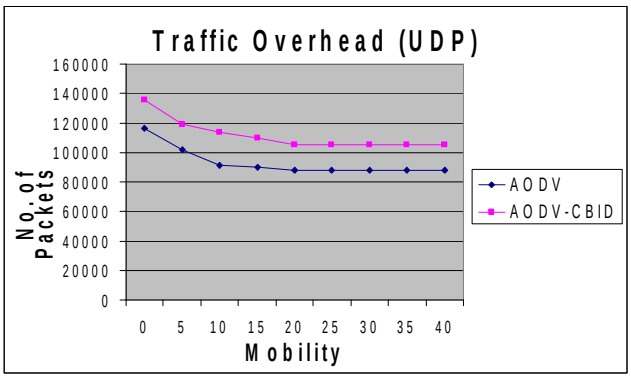

Fig 3: Traffic Overhead for UDP with Low Load

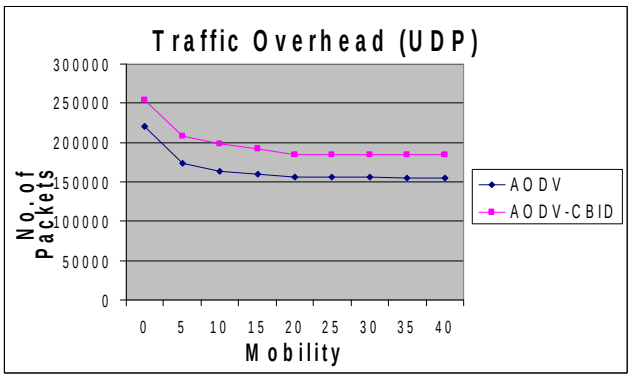

Fig 4: Traffic Overhead for UDP with Medium Load

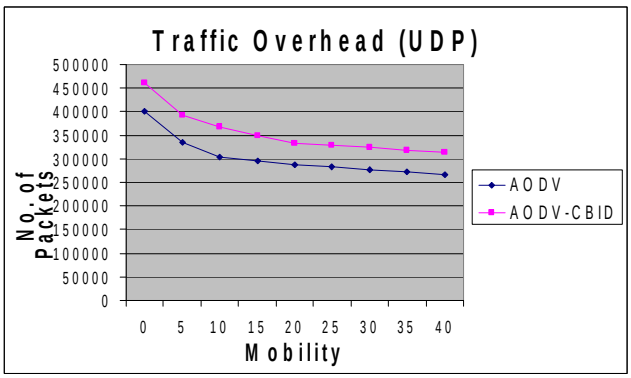

Fig 5: Traffic Overhead for UDP with High Load

Figure 3 to 5 depicts the overhead caused by the use of CBID with AODV as a routing protocol for UDP traffic. It can be observed that the no of packets transacted during simulation is decreasing as the nodes become more mobile.

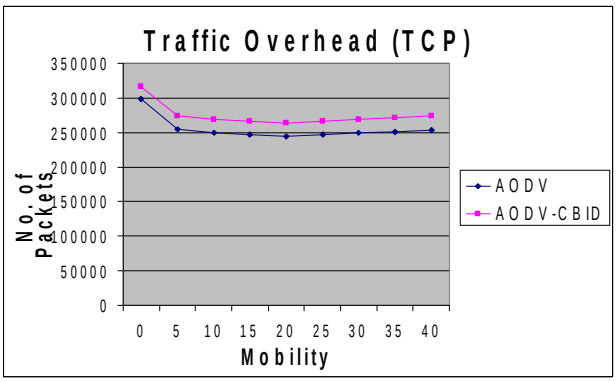

Fig 6: Traffic Overhead for TCP with Low Load

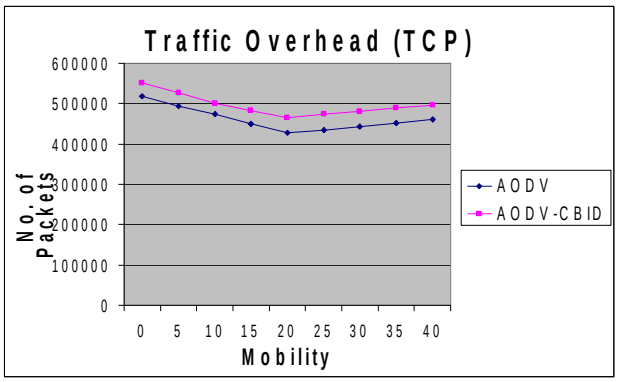

Fig 7 Traffic Overhead for TCP with Medium Load

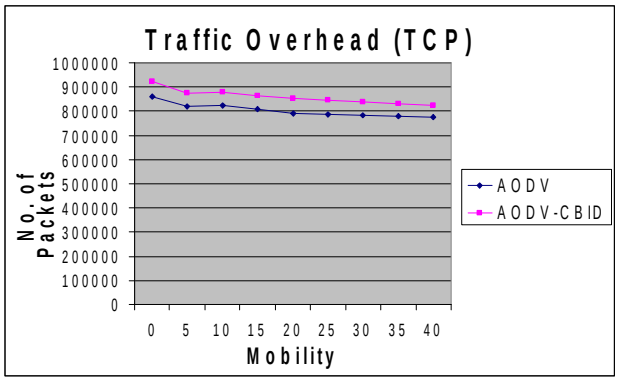

Fig 8: Traffic Overhead for TCP with High Load

Figure 6 to 8 depicts the overhead caused by the use of CBID with AODV as a routing protocol for TCP traffic. The CBID causes low overhead in case of TCP traffic than UDP traffic.

6. Conclusion

In this paper, a comparative analysis of a generalized Intrusion Detection Clustering Scheme for ad-hoc networks is carried out. The CBID scheme was found to be simple and offers low over head in terms of memory usage and number of messages exchange. The fast, efficient and fair election process for selection of monitoring node proposed in CBID has reduced the number of packets exchanged for cluster formation and intrusion detection. The scheme uses cooperative approach to coordinate among different nodes for intrusion detection and prevention against different attacks within the ad-hoc network.

The effectiveness of the CBID protocol in comparison to existing methods is tested under different types of traffic, mobility and load conditions. Also, the overhead of CBID protocol is measured and is turned out to be uniform, irrespective of the mobility and route formation.

7. Future Work

Currently we are investigating on various issues to make CBID more effective, secure and efficient. The issues under consideration includes devising a certain mechanism for willingness criteria [14] of the voting rather than using random numbers and trust level calculation. Cooperation between route formations with the trust level to avoid formation of a route with detected malicious node. Implementation of distributed gateway (D-GW) concept for distant head nodes (HD) that are more than 2-hop apart [15]. Studying the effect of malicious gateway node and making the communication of head node more secure for collaborative intrusion detection. Moreover evaluation of CBID using different routing protocol is required to check its effectiveness and independence.

8. References

[1] P. Albers, O. Camp, J.-M. Percher, B. Jouga, L. Mé, R. Puttini, “Security in Ad Hoc Networks: a General Intrusion Detection Architecture Enhancing Trust Based Approaches”, in Proceedings of the First International Workshop on Wireless Information Systems (WIS-2002), Apr, 2002.

[2] Yongguang Zhang, Wenke Lee, “Intrusion Detection in Wireless Ad-Hoc Networks” , Proceedings of the Sixth Annual International Conference on Mobile Computing and Networking, MobiCom 2000, Boston, Massachusetts, Aug 6 11, 2000, pp 275-283.

[3] Chunsheng Li, Qingfeng Song, Chengqi Zhang: “MA-IDS Architecture for Distributed Intrusion Detection using Mobile Agents”, Proceedings of the 2nd International Conference on Information Technology for Application (ICITA), 2004.

[4] Yi-an Huang, Wenke Lee, “A Cooperative Intrusion Detection System for Ad Hoc Networks”, in Proceedings of the 1st ACM workshop on Security of ad hoc and sensor networks (SASN), Fairfax, Virginia, October 31, 2003.

[5] Kashan Samad, Ejaz Ahmed, Waqar Mahmood, “Simplified Clustering Scheme for Intrusion Detection in Mobile Ad Hoc Networks”, 13th International Conference on Software, Telecommunications and Computer Networks (SoftCOM), Split, Croatia, September 15-17, 2005.

[6] S. Banerjee, S. Khuller, “A Clustering Scheme for Hierarchical Control in Wireless Networks”, in Proceedings of IEEE INFOCOM, 2001

[7] Mingliang Jiang, Jinyang Li, Y.C. Tay: “Cluster Based Routing Protocol (CBRP)”, Internet Draft, Jul, 1999.

[8] P. Krishna, N. H. Vaidya, M. Chatterjee, D. K. Pradhan: “A cluster-based approach for routing in dynamic networks”, ACM SIGCOMM Computer Communication Review, 27(2):49-64, 1997.

[9] FH Wai, YN Aye, NH James, “Intrusion Detection in Wireless Ad-Hoc Networks”, 2003.

[10] Yongguang Zhang, Wenke Lee, Yi-An Huang, “Intrusion detection techniques for mobile wireless networks”, Wireless Networks, v.9 n.5, p.545-556, September 2003.

[11] “Intrusion Detection Systems (IDS) Part I - (network intrusions; attack symptoms; IDS tasks; and IDS architecture)”, Jun 14, 2004.

[12] Dorothy E. Denning, “ An intrusion-detection model. IEEE Transactions on software engineering”, February 1987.

[13] Yu Liuy, Yang Liy, Hong Many, “MAC Layer Anomaly Detection in Ad Hoc Networks”, 6th IEEE Information Assurance Workshop, USA, 15-17 June 2005

[14] Arvind Ramalingam, Sundarpremkumar Subramani, Karthik Perumalsamy, “Associativity based cluster formation and cluster management in ad hoc networks”.

[15] Yunjung Yi, Mario Gerla, Taek-Jin Kwon , “Efficient Flooding in Ad-Hoc Networks using On-Demand (passive) Cluster Formation” , in Proceedings of Mobihoc, June, 2003.